DOCUMENT RESUME

ED 255 562 TM 850 205

AUTHOR Murphy, Patricia A.
TITLE A Model for Improving Assessment Techniques Used in Courseware Production: Proposal for Change in Courseware Production Team Structure.
PUB DATE Apr 85
NOTE 35p.; Paper presented at the Annual Meeting of the National Council of Measurement in Education (Chicago, IL, April, 1985).
PUB TYPE Speeches/Conference Papers (150) -- Reports - Descriptive (141)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; *Computer Assisted Testing; *Courseware; Models; Programed Instructional Materials; Programing; Research and Development; *Test Construction
IDENTIFIERS *Instructional Systems Development

ABSTRACT

The videodisc, audiodisc, overlapping instruction, mapping, adaptive testing strategies, intricate instructional branching, and complex media selection models are common expectations for quality courseware today. A two team development structure, consisting of an instructional development team and a test development team, is proposed. This paper presents a flexible test construction process model and illustrates its implementation with three prominent courseware design and development models: (1) the MicroTICCIT system of the Hazeltine Corporation; (2) the Control Data Corporation's model for PLATO as described in their CREATE curriculum; and (3) the Authority model by Interactive Training Systems, Inc. The two team approach allows for greater skill specialization for both test personnel (with a measurement background) and courseware developers (with a learning theory or curriculum development background). Across project application of test personnel can also save time and money. Flowcharts illustrate the test construction model and the three courseware development models. Ten references are included. (BS)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED255562

A Model for Improving Assessment Techniques Used in Courseware Production: Proposal for Change in Courseware Production Team Structure

Patricia A. Murphy

ManTech International
1006 West Street
Laurel, MD 20707
and
University of Maryland

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☒ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

P. A. Murphy

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

TM 850 205

Annual Meeting of National Council of Measurement in Education, Chicago, IL. April, 1985.

A Model for Improving Assessment Techniques used in Courseware Production: Proposal for Change in Courseware Production Team Structure

Introduction. As software enters the classroom and the training room, a new skill category has opened up. The courseware developer must be a person with a new set of skills, packaged in a manner seen as unique in the marketplace. The package of skills of the courseware developer includes the ability to apply learning theory and curriculum development theory to the design and development of instructional components, communicate effectively using audio and visual media, write the instructional components using a programming or authoring language, and frequently, speak in the language of the subject matter expert.

In addition to instructional sequences, the course developer must also plan the assessment strategy. The additional abilities here include the application of measurement theory to the development of test items, testing sequences, and the testing session. The course developer writes tables of specifications, constructs test items, oversees the content validation of the items, and constructs the test, even to the point of establishing parallel or nominally parallel test forms. Beyond, this the course developer also must establish a branching strategy for the courseware based upon assessment needs and implement the rules to affect that strategy.

The new techniques for branching based upon the alternative selected in an embedded item need to be understood and implemented by the course developer. The data collection procedures are also implemented by the course developer so the validation of the course can be completed at the close of the pilot testing. In the validation phase, the course developer is expected to use the appropriate types of analyses, both quantitative and qualitative, to evaluate the courseware.

Problem. In order to improve the quality of assessment available in courseware, the suggestion being made today is for the separation of the courseware team into two groups, instructional developers and test developers. This separation will allow a model for test construction to stand apart from the model for the development of instruction. The model for test construction must support the development model, as well as retain the integrity of the test construction process.

Industrial training and publishing firms have structures that currently employ courseware developers in the same manner that trainers' or teachers' jobs were previously defined (Bork, 1979). That is, these people were expected to be knowledgeable about how to instruct as well as how to assess. With the advent of so many new techniques in courseware development, it is no longer reasonable to expect one individual to be versed in all areas that must come together to make quality courseware. The videodisc, audiodisc, overlapping instruction,

mapping, adaptive testing strategies, intricate instructional branching, and complex media selection models are common expectations of quality courseware today, and therefore, expectations of its author during production.

Using technology for the purposes of educating, demands cost-efficient and instructionally intense courseware. Developed, effective techniques should be shared and used repeatedly in appropriate situations. While individuals will develop expertise in some aspects of courseware production, they will not be as effective in other aspects. The broad range of required skills and the ever evolving technology demand specialization.

Instead of lowering the standard for courseware, establish the development team structure so it is supportive of the needs of the whole courseware process. To meet this proposed change in the structure of the team, a reanalysis of the tasks of the team members is required. This reanalysis will enable the building of a model of a test construction process that is flexible and can be melded with varied courseware development process models. This model of test construction shall support the claim that two separate teams with different skills sets are needed to effectively and efficiently produce courseware. The model of test construction should establish a test construction process that allows the most refined testing techniques to be implememted by professionals interested in quality assessment within courseware.

Division of Skills. The division of labor for complete courseware development separates into the development of instruction and the construction of tests. The development of instruction includes learning hierarchies, course, unit, and lesson maps, lesson specifications, scripting and storyboarding authoring, and lesson integration. The construction of tests includes testing strategy layout, table of specification, content domains, item construction, item content validation, embedded item format or test design, item and test scoring strategy, within test branching strategies and authoring and data collection procedures and implementation.

Two roles break out of the tasks identified under the development of instruction. They are the instructional designer and the courseware developer. Similar roles would break out under the tasks for test construction; however, it has been my experience that testing is marching forward if it has a single member on a development team. In an effort to see the model implemented and not thrown out because of costliness, the test team presented here encompasses test developers and measurement specialists as a single group with both sets of skills.

The tasks to be completed in the development of courseware are:

By Test Developers and Measurement Specialists

- o Determine assessment strategy and techniques
- o Determine appropriate item types based upon objectives
- o Specify content domains for each objective or group of objectives
- o Develop table of specifications
- o Review existing test material
- o Construct pools of items or self-contained, branching scenarios
- o Specify graphics, video, and audio necessary during assessment
- o Conduct a review and revision of the items and scenarios
- o Correlate subject matter experts' ratings for item difficulty and content validity prior to item trial
- o Conduct a preview of the items
- o Establish data collection procedures for tests and test items within the computer management capabilities
- o Design test layout to include provisions for parallel forms where necessary, and adaptive testing where designed
- o Implement test layout on instructional delivery medium
- o Implement scoring mechanism for the stated criteria
- o Debug testing sequences and sessions
- o Conduct a preview of the tests and testing sequences
- o Prepare instructor test manuals
- o Apply appropriate analysis techniques following pilot testing (both quantitative and qualitative)

- o Provide a quantitative measure of common error analysis
- o Base revision of items, tests, and testing sequences on analysis of quantitative and qualitative information

By Instructional Designers and Courseware Developers:

Instructional Designers

- o Assess learner characteristics
- o Analyze the structural characteristics of content
- o Analyze the characteristics of the learning environment
- o Review the learning objectives
- o Sequence learner outcomes
- o Specify instructional strategies
- o Sequence learner activities
- o Determine media selection for instructional activities
- o Prepare lesson specifications for courseware production

Courseware Developers

- o Prepare storyboards or grids based upon lesson specifications
- o Author lesson based upon lesson specification to include scripts for audio or video
- o Write textual training materials for independent study and traditional classroom activities where designed
- o Debug CAI, audio, and video lessons
- o Revise materials and courseware
- o Conduct technical review

- o Conduct instructional review
- o Oversee shooting and editing of videos and audios needed during instruction
- o Integrate lesson into units; add transitional frames where necessary
- o Prepare support materials including documentation and management documents
- o Administer pre-pilot and validation phases

Coordination between Teams. This separation by skills allows greater specialization by team members. The test developers are people with a measurement background. The courseware designers and instructional developers are people with a background in learning theory or curriculum development. All team members must have a working knowledge of the authoring language, the video and audio systems, and the techniques for mixing the media types.

The separation of skills assures that each courseware product will be developed by individuals with the skills to complete a quality product. But allowing greater specialization of skills does force greater coordination between the two teams. This coordination is necessary to ensure quality courseware and is the additional requirement placed on the teams because of the separation.

The areas of coordination are:

- o Color and format conventions used in the courseware
- o Assessment strategies in response to the needs of the development team's outline of instruction
- o Analysis techniques and reporting with the quality assurance groups and courseware production groups
- o Expertise on the skills of item writing to developers working on practice items
- o Criteria used in branching strategies for instruction and assessment
- o Staff training for users of the courseware

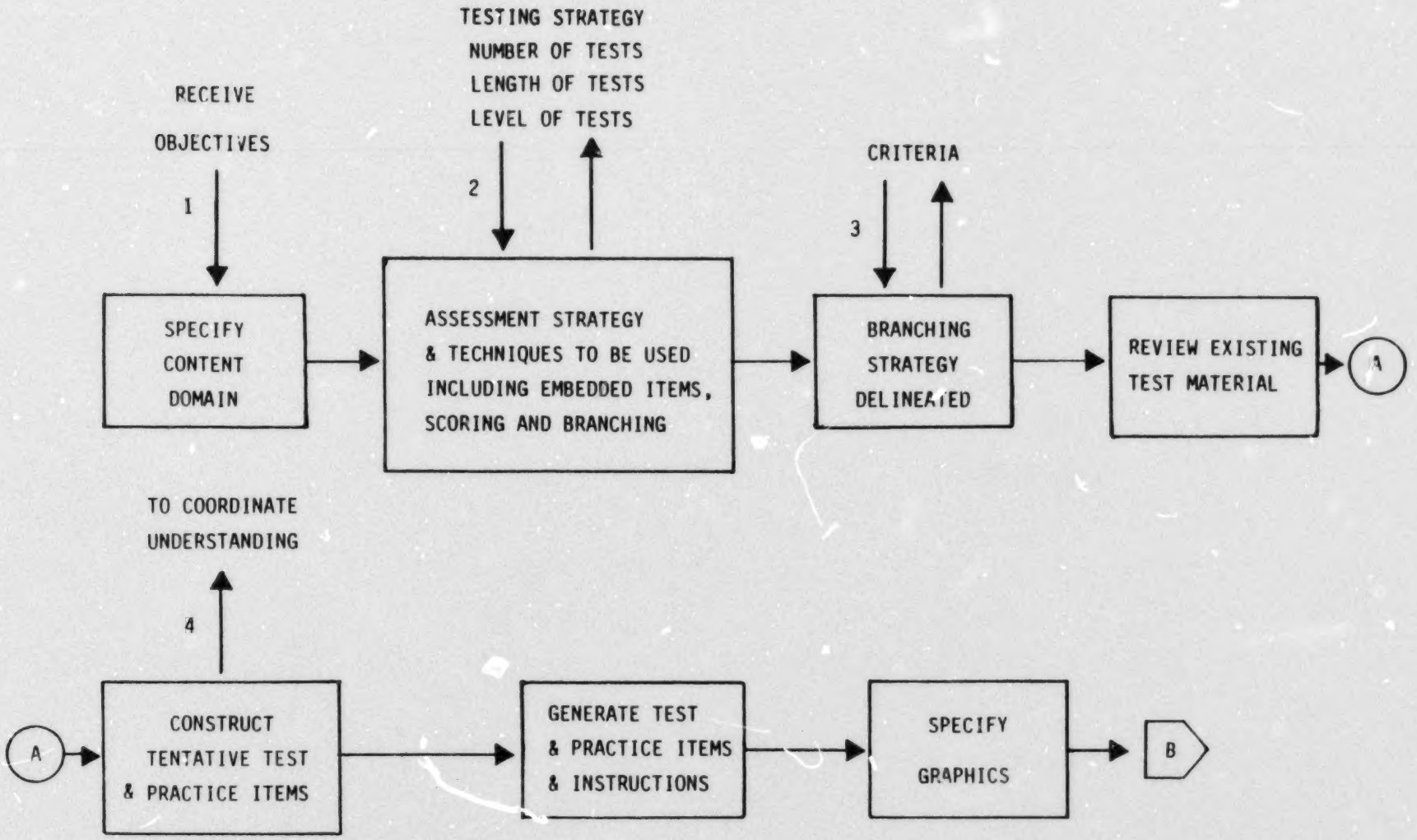
A discussion of the areas of coordination is accompanied with the proposed model of test construction. Following this, the flexibility of the model will be demonstrated.

Test Construction Model.

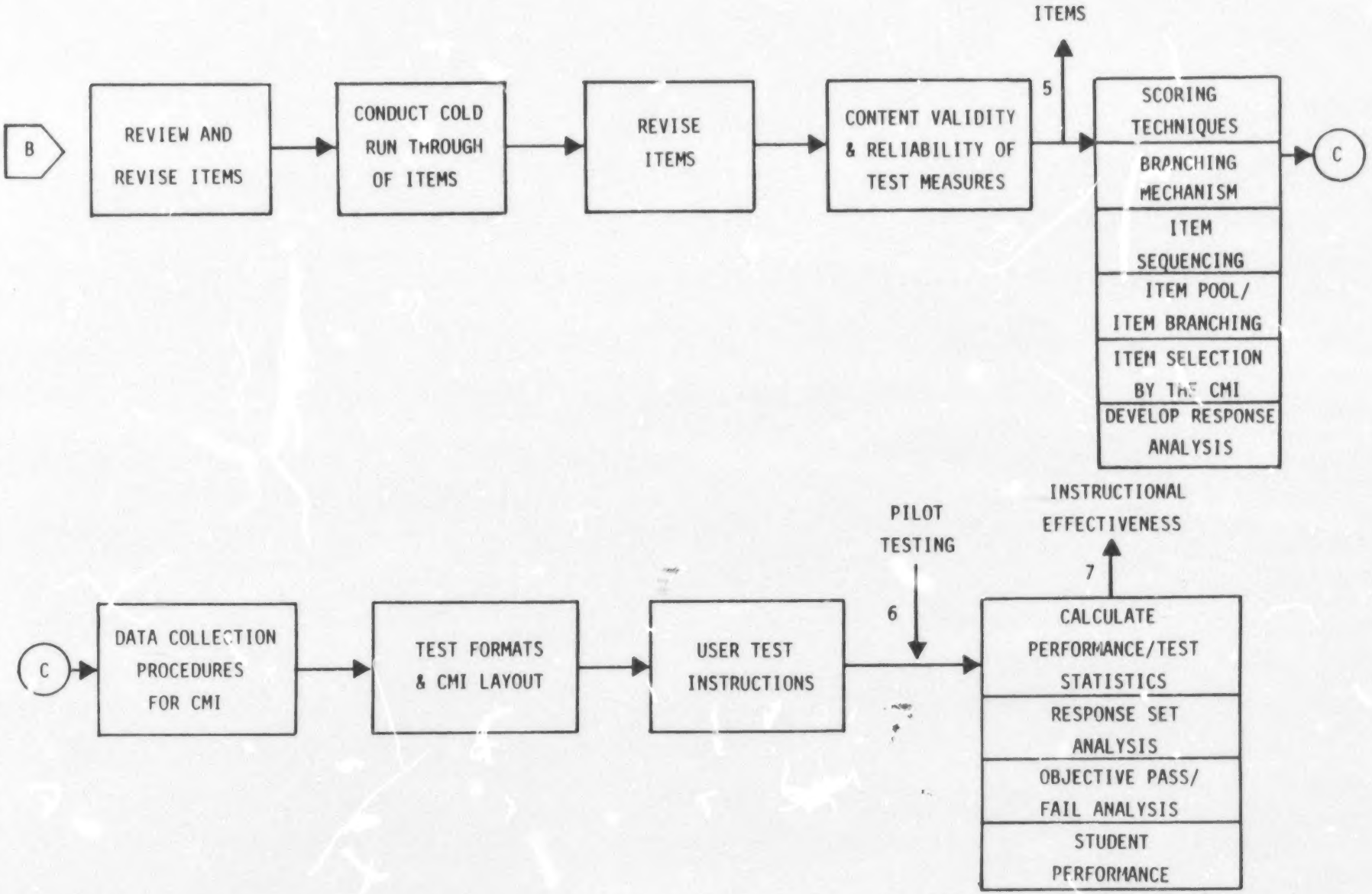
(Graphic 1)

The test construction model being suggested is diagrammed. The vertical arrows indicate points of coordination between the two teams.

The first place the coordination occurs is in determining the CAI color and format conventions to be used in the courseware. The two teams meet and agree on these standards for the course if they have not previously been established.



TEST CONSTRUCTION MODEL



TEST CONSTRUCTION MODEL - Page 2

The test developers begin their process following the completion of course, unit, and lesson maps. They base their testing strategy on the structure of the course established by the courseware designer. The next point for coordination is upon completion of the construction and content validation of items. The test developers pass their items to the team of courseware developers. The courseware developers in turn, may add practice items to the item pool. This is encouraged, especially if items of lesser difficulty are needed to enhance learning. This early delivery of the items ensures that instruction and testing interpret the learning objectives in the same manner. If the instructional designers have difficulty with any of the items, a dialogue between the two teams follows. The sameness in interpretation is necessary if the separation of production into two development teams is not to hinder effective courseware production.

The next area for coordination is branching. The conditions and criteria to be used in the branching strategies should be agreed upon by both teams and used consistently. The test developer shall use this strategy in item and test scoring, branching and data collection procedures. The courseware developers shall use this strategy to author the branching mechanisms between tests and instruction and to determine remediation loops.

Prior to scripting and storyboarding the lesson, the use of feedback in the course of instruction must be agreed upon by the teams. The philosophy underlying feedback and whether or not to make the feedback responses uniform throughout are decisions to be made through consensus of both teams.

The actual test layout is constructed by the test personnel; however, once the tests are designed, the test must be entered into the system. The courseware developer needs to review the tests to be sure that the practice items and the test items are coordinated. This ensures that the student will be assessed on the skills or knowledge that was taught and practiced. This also provides a time for the courseware developer to view the final test items prior to finalizing the instructional components. The timing of this final exchange is important. The courseware developer should not be hindered in the progress of authoring because of a lack of knowledge about tests. This is an important feature of the flexible CAI test construction model.

The test construction model also encourages an exchange between courseware developers and test developers following the pilot testing. The purpose of this exchange is to base the revision of the courseware on the results of the pilot testing .

The testing team carries out the analysis of the student data from the pilot testing phase. This analysis includes tracing student behavior on test items back to the objectives and instructional components supporting the objective. This tracing serves as a means of pinpointing

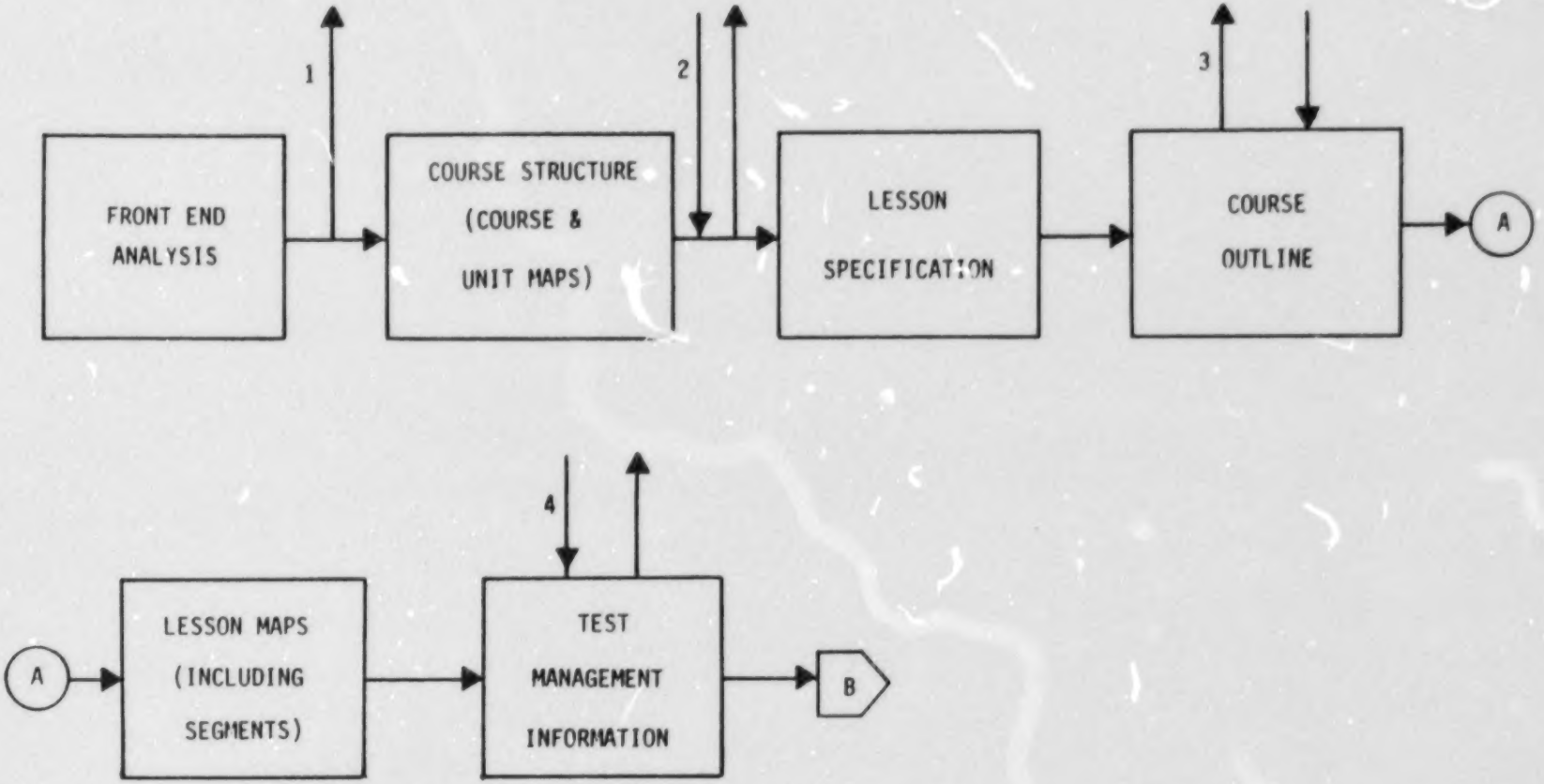
the areas within the courseware where the instruction was not effective. If the course standards are criterion-referenced, and particularly if they involve the mastery of instruction, pinpointing the problem areas in the instruction is necessary for the revision to increase the effectiveness of the courseware. The testing team supports the instructional designers and courseware developers in making revisions by providing the information on particular instructional components.

The rest of this discussion shall demonstrate the flexibility of the model. The flexibility shall be apparent by fitting this model with three prominent courseware design and development models. The testing portion of these models allows only for elementary testing techniques to be implemented. The test construction model proposed for developing computer-assisted testing sessions within courseware builds in the time for richer techniques to be implemented.

Implementation One. Consider the model of development suggested by Hazeltine Corporation and their MicroTICCIT system (Hazeltine Corp.,1983).

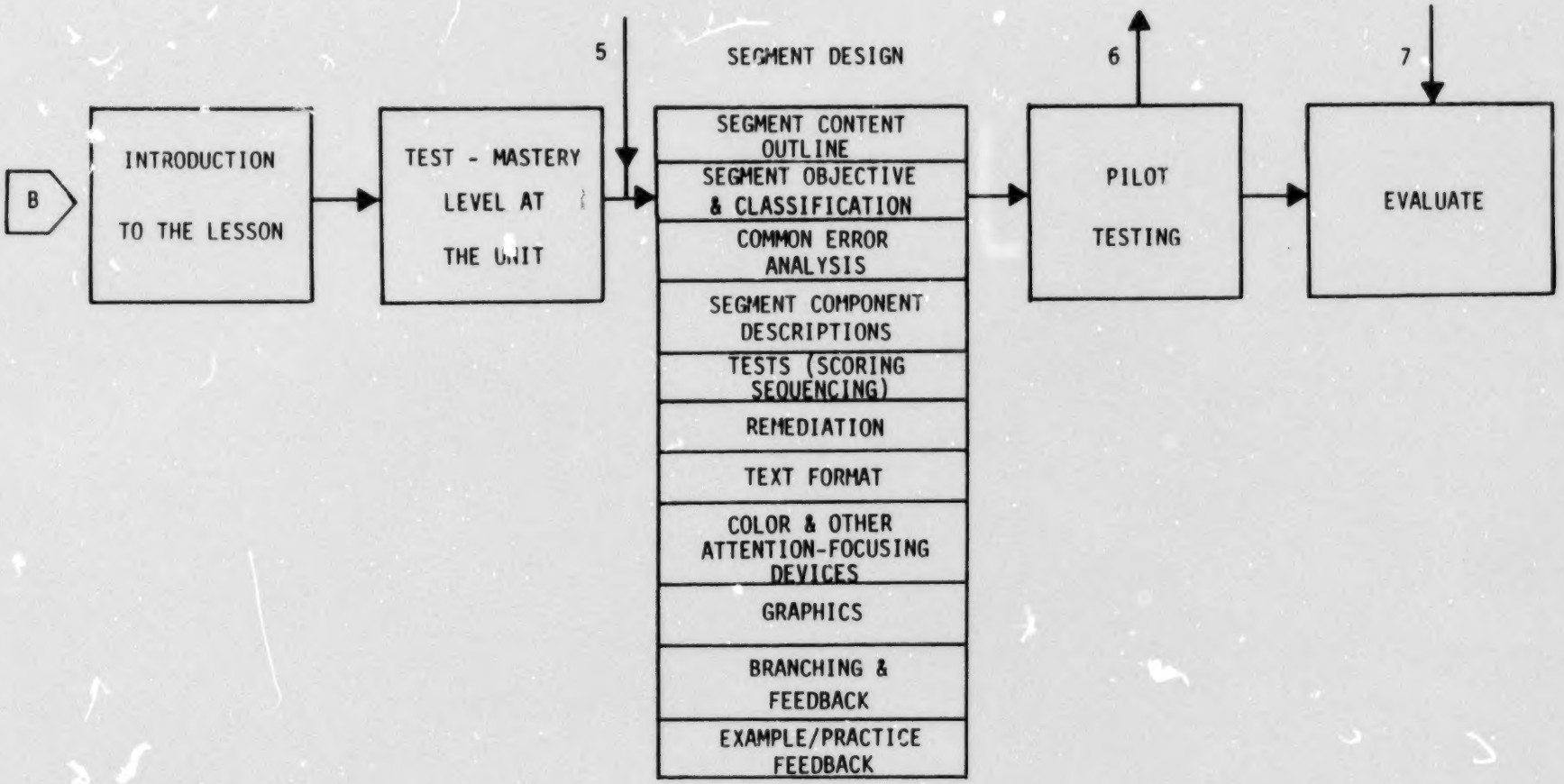
(Graphic 2)

Hazeltine suggests front-end analysis be done, followed by the clustering of objectives into course, unit, lesson, and segment levels. The Lesson Specification Document developed next describes the detail of the lesson and notes any agreed-upon standards for format and color. All lesson prerequisites, including facts, concepts, rules, and



COURSE DESIGN

Hazeltine Corp. MicroTICCIT



COURSE DESIGN

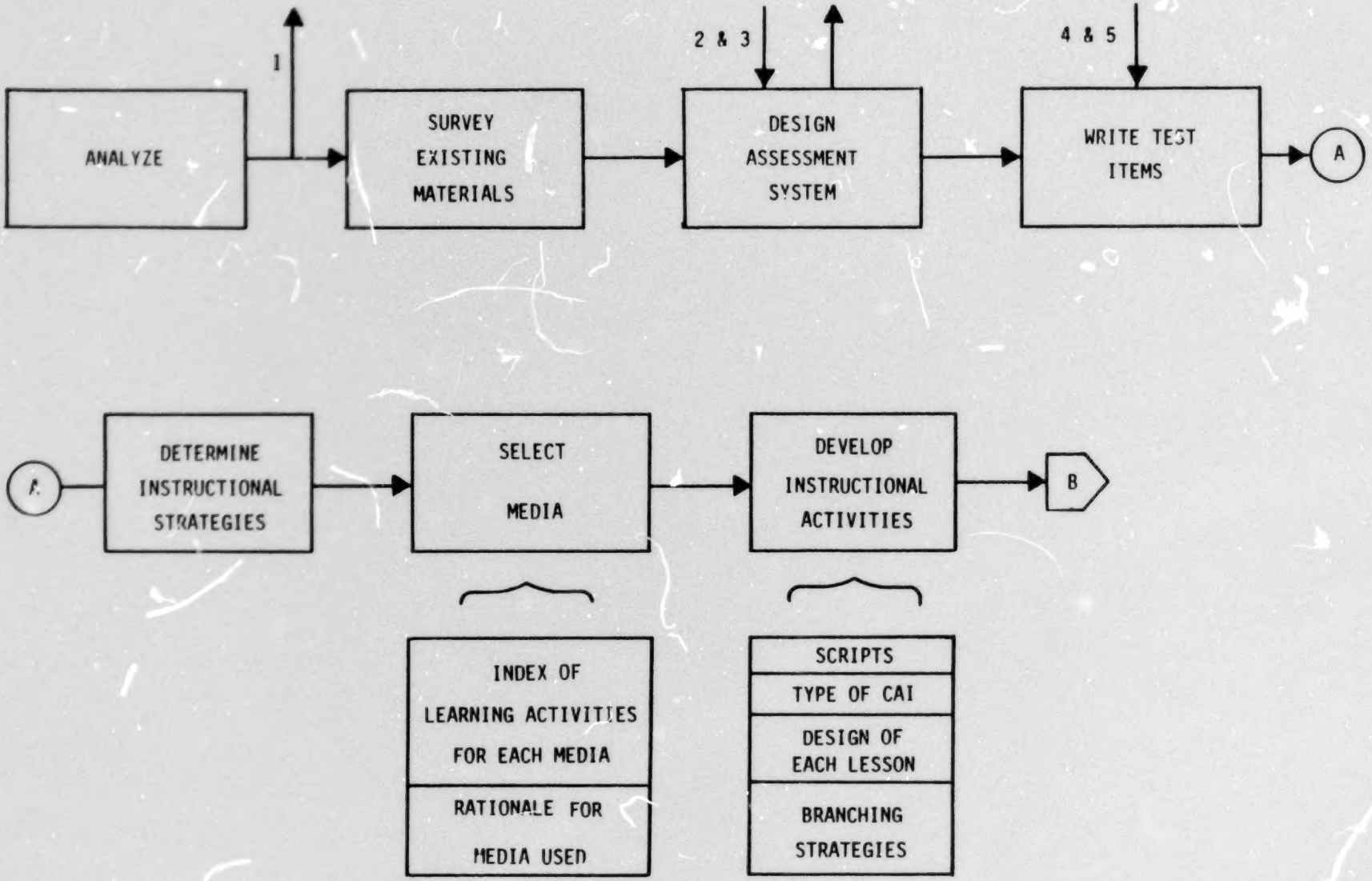
procedures, are listed. Enabling objectives are part of the segment specification, while terminal objectives are associated with the lesson level.

Authoring, now done in the Adapt language, begins with the lesson introduction. The lesson introduction may be used in the roles of motivation, summation, or transition. Tests for a lesson are given at the end of the instructional lesson. The enabling and terminal objectives of the lesson are tested together. The unit level tests are generally used for testing mastery. The actual storyboarding occurs at the segment level. A segment is similar to an instructional component. MicroTICCIT authors for CAI, video, and audio media. Hazeltine suggests a mix of these media depending upon content.

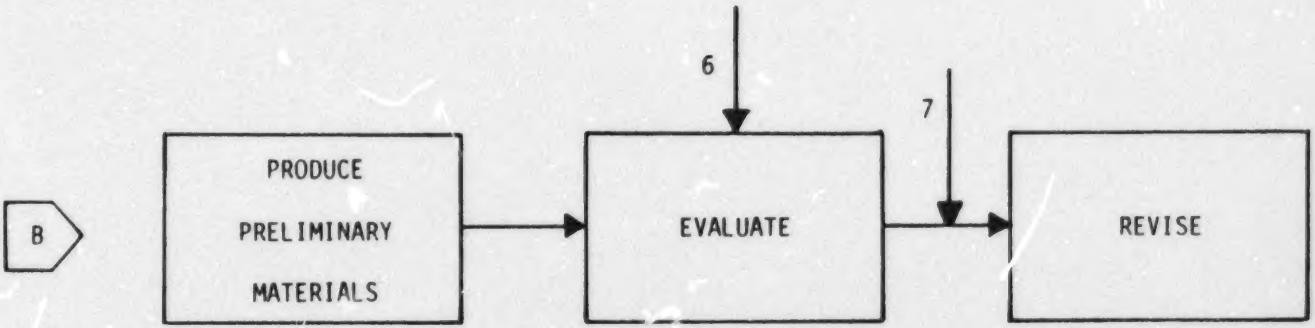
The arrows indicate the areas of coordination between the instructional team and the test team. The test construction model adapts to the Hazeltine proposed courseware production model. It does not violate any of the prescribed steps or stages. The test construction model also remains intact during this adaptation. The areas of coordination remain stable. What changes is the time between the areas of coordination.

Implementation Two. The next model of development is for the Control Data Corp.'s model for PLATO as described in their CREATE (Control Data Corp., 1978) curriculum. This model uses an instructional systems design (ISD) basis.

(Graphic 3)



PLATO CONTROL DATA
INSTRUCTIONAL SYSTEMS DEVELOPMENT BASIS



PLATO CONTROL DATA

INSTRUCTIONAL SYSTEMS DEVELOPMENT BASIS

Page 2

The development phase of the process here begins with a survey of existing materials that were developed for similar content. The useful information from the survey is used throughout the development phase. This process model requires assessment to be determined at the start of development. This is consistent with the ISD practice of teaching to the test. After all the test items and branching criteria are written, the development of instruction then begins. The courseware developers have the test items, the tests, and the branching criteria available to them throughout development. The media are not selected until all the learning activities for each instructional strategy has been determined. The media selected on the basis of the needed capability to successfully implement the learning activity. The same is true for the medium of each test item. It is selected only after the item is written and the need established.

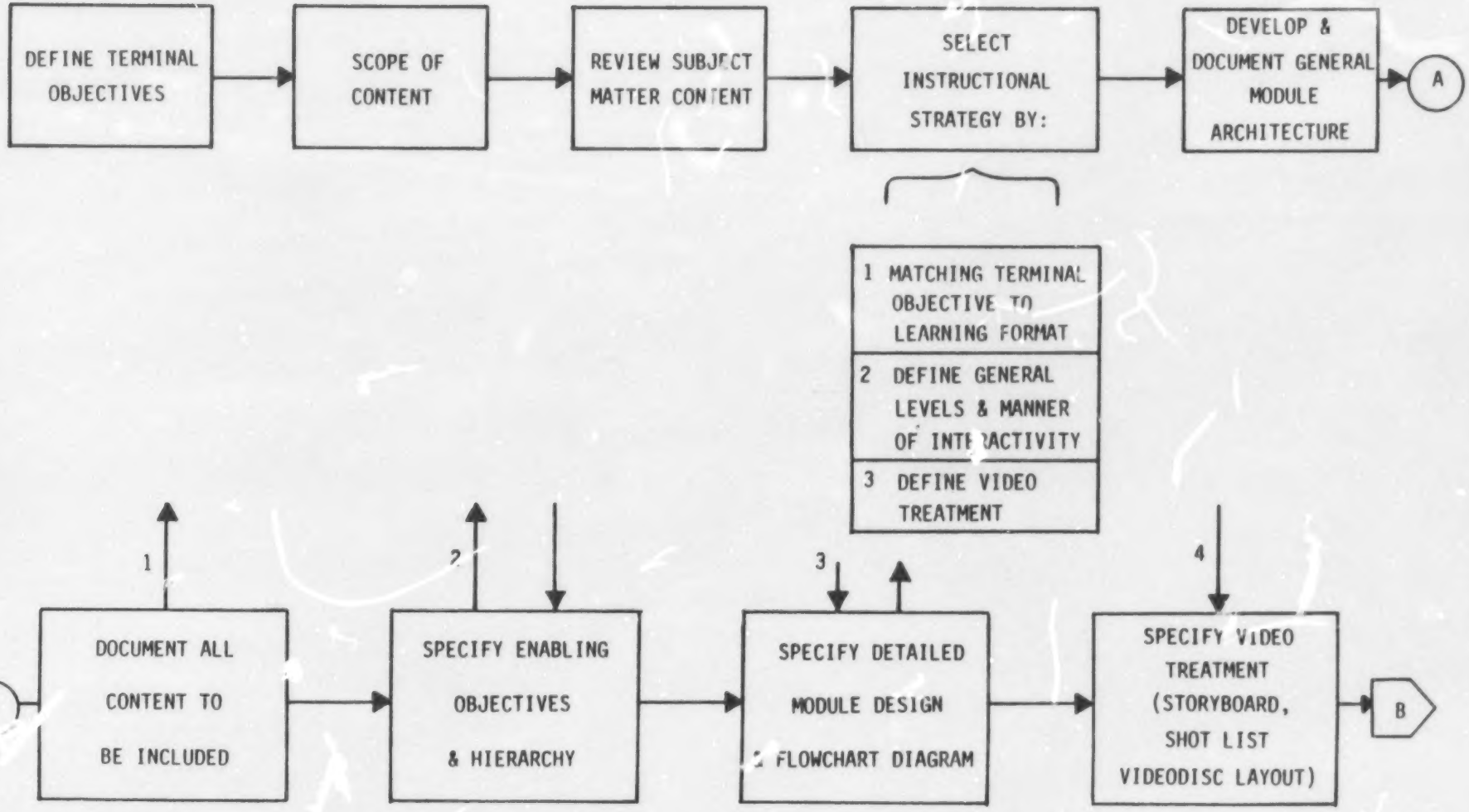
Scripts are then written for each learning activity according to the chosen medium. The design of each individual lesson includes the branching strategies employed internal and external to the lesson. The external branches are concerned with missed prerequisites, forced lesson order, remediation loops, and superior performance. After all the details of the lesson are determined, preliminary CAI, video, and audio materials are developed. An evaluation is recommended, followed by a revision of the materials.

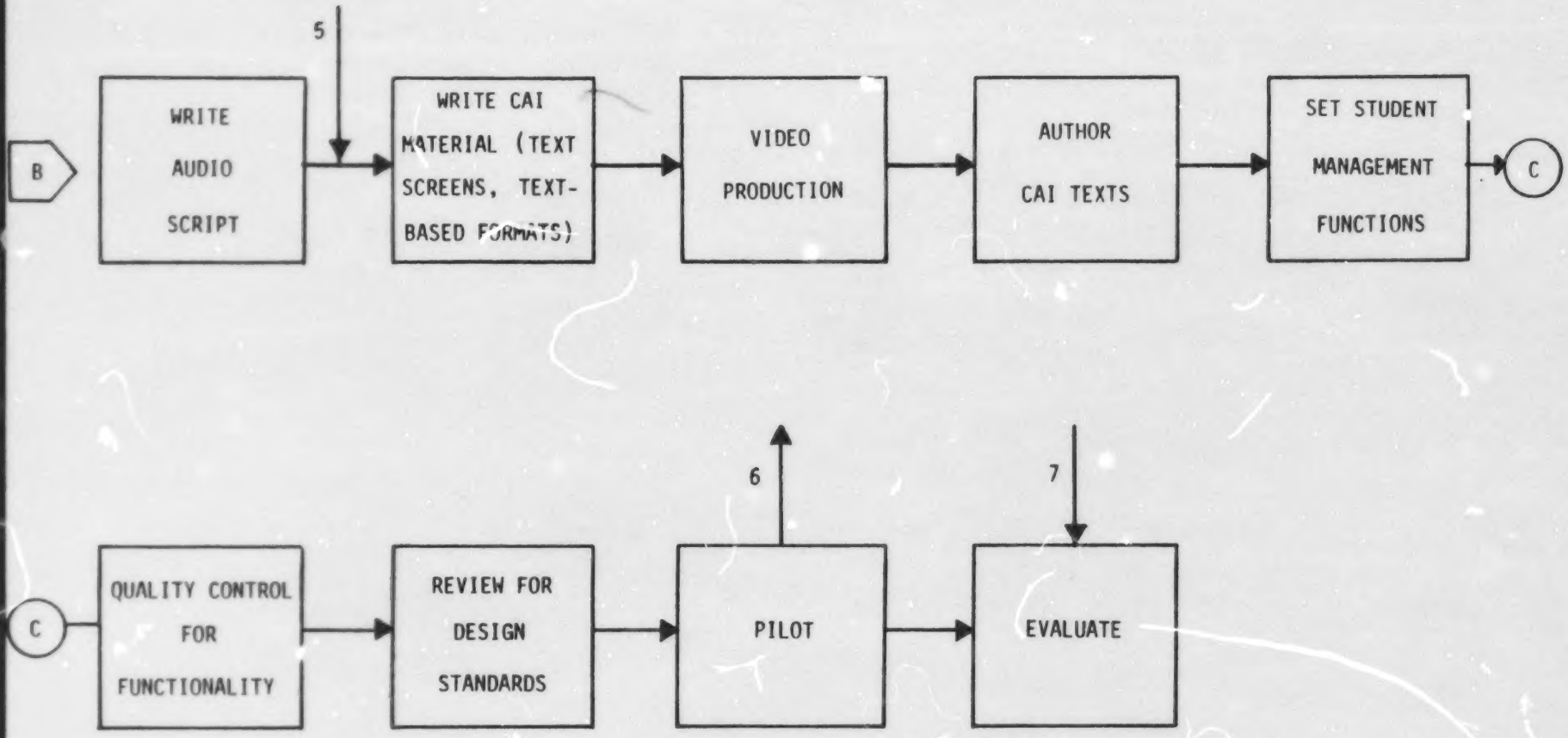
Again, the arrows on the diagram of the model represent areas of coordination. Again the position of the testing process in the model

must shift. But there is nothing prohibiting the use of more advanced testing techniques that the PLATO model originally suggested. This adaptation of the CAI test construction model requires parallel development. Tests and instruction need to be authored simultaneously for this design to be realistic in its time constraints. Because of the skills separation of the test personnel and the instructional team, this does not cause a problem.

Implementation Three. There are several other competing models of courseware development. Steinberg (1984) and Landa (1984) both have models of significant value. And The Department of Defense's adaptation of ISD, entitled," Interservice Procedures for Instructional Systems Development," (1975), also has a feasible plan for implementation. The model to be shown for implementation three however, has several unique features that demonstrate a creative approach to courseware design as well as the flexibility of the test construction model.

The development model was shaped by Lasker and Lubin, formerly with Harvard University. This model is now implemented under the name of Authority by Interactive Training Systems, Inc. The model is unique in the way it helps to structure learning. Authority uses built in learning formats, as the authors had developed for "Sesame Street" and "Electric Company." Lasker and Lubin developed learning formats which prestructure the learning strategy. Only the learning activity needs to be specified by the developer based upon the learning format. (Graphic 4)


DEFINE TERMINAL OBJECTIVES
SCOPE OF CONTENT
REVIEW SUBJECT MATTER CONTENT
SELECT INSTRUCTIONAL STRATEGY BY:
DEVELOP & DOCUMENT GENERAL MODULE ARCHITECTURE
A
1 MATCHING TERMINAL OBJECTIVE TO LEARNING FORMAT
2 DEFINE GENERAL LEVELS & MANNER OF INTERACTIVITY
3 DEFINE VIDEO TREATMENT
1
2
3
4
A
DOCUMENT ALL CONTENT TO BE INCLUDED
SPECIFY ENABLING OBJECTIVES & HIERARCHY
SPECIFY DETAILED MODULE DESIGN & FLOWCHART DIAGRAM
SPECIFY VIDEO TREATMENT (STORYBOARD, SHOT LIST VIDEODISC LAYOUT)
B



INTERACTIVE TRAINING SYSTEMS, INC.

AUTHORITY

The prestructuring helps to focus the developer on the issue of relating the content to the audience. The question of how to structure a lesson or grab attention are answered once the learning format is chosen. Authority is a videodisc with an audio based system. It does have CAI text pages and CAI overlay pages, but these are not meant to be the main media of instruction. Testing can be done using traditional multiple-choice questions, or a more dynamic testing session can be done through interaction between the student and the computer using videodisc presented testing scenarios.

The development model by Interactive Training Systems begins on an upper level. The terminal objectives are identified and the scope of the content decided. An instructional strategy is selected by matching terminal objectives to a prestructured learning format. The level of interactivity is defined, and the video treatment decided upon. All this information is used to develop the general module architecture. Then the development process drops a level and specifies the enabling objectives to be taught and their sequence. The detailed module design and flowchart follow. The video treatment, the audio script, and the CAI materials are specified. Video production is completed prior to authoring CAI texts. All management functions are set following the completion of all authoring functions. A quality control check is done on the functionality of the module as designed. The module is reviewed and pilot tested. Again the arrows on the diagram indicate areas of coordination. Because how content is taught influences how it should

be tested. The test groups play a more active role in the specification of objectives in this model. The learning formats are partially chosen because of the kind of behavior required in the objective. The manner of testing the objective will correspond with the learning format and be in compliance with the objective. The positioning of testing process blocks occurs in clusters. But there is a necessity for parallel development and the separate teams will aid in keeping linear development time down.

Conclusion. Using a separate testing model allows the testing process to retain its integrity. The best means for implementing a separate testing process is through a separate test team. This separation allows greater specialization of skills for both test personnel and courseware developers.

Because of the frequency across development models that parallel development and idle time ordinarily occur, this separation of skills is most useful where more than one courseware development project is occurring. If the courses or modules are out of phase, a lower total cost is incurred because of the lower size of the labor force required and the application of the same sets of skills across projects. This across-project application of test personnel is significantly more pronounced when item pools are available for the content of the course.

To increase the value of the two team approach, specialization needs to be fostered. A greater level of expertise in an area will develop when the "experts" are sharing the techniques, algorithms, and creative ideas they develop and use. The exchange of knowledge should not happen accidentally, but purposefully. The exchange should be structured around tactics and techniques.

References

Avner, Allen. Smith, Stanley and Tenczar, Paul. (1984). CBI authoring tools: Effects on productivity and quality. Journal of Computer-Based Instruction. (3) 85-89.

Bork, Alfred. (1979). Interactive learning. American Journal of Physics, 47 (1).

Control Data Education Company. (1978). CREATE. MA: Author.

Hazeltine Corporation. (1983). Courseware Design. Reston, VA: Author.

Interactive Training Systems Inc. (1984). Interactive Videodisc Module Development Tasks. Cambridge, MA: Author.

Jackson, Rex. (1970). Developing Criterion-Referenced Tests. Princeton, N.J : Educational Testing Service.

Landa, Ruth K. (1984). Creating Courseware. New York, N.Y. : Harper and Row Publishers.

Lathrop, Ann, and Goodson, Bobby. (1983). Courseware in the Classroom. Menlo Park, CA : Addison-Wesley Publishing Co.

Steely, Don. (1983). Instructional design and CAI in Run: Computer Education. Harper, Dennis. and Stewart, James H. (Eds.) Montery, CA. : Brooks/Cole Publishing Co. 126-128.

Steinberg, Ester R. (1984). Teaching Computer to Teach. Hilldale, N.J. :L. Erlbaum Associates.